

Currículo de Kodu: Teclado y Ratón

Tabla de Contenidos

Currículo de Kodu: Teclado y Ratón	3
Enseñando con Kodu	4
Alcance y Secuencia del Currículo de Kodu	6
Sesión 1: Navegar, Introducción a Conceptos de programación, Adicionar objetos...	7
Hoja del Estudiante Actividad 1: Comer Manzanas	7
Sesión 2: Crear un Paisaje.....	8
Hoja de Estudiantes - Actividad 2: Hacer Paisajes	13
Sesión 3: Usar el Controlador para Mover Personajes, Crear Rutas, y Establecer Comportamientos	14
Hoja de Estudiantes – Actividad 3: Comportamiento del Objeto y Rutas	19
Sesión 4: Hacer Clones y “Creatables” (Seres que se pueden crear)	20
Hoja del Estudiante Actividad 4: Personajes y Argumento.....	23
Actividad de Desafío.....	24
Actividad Suplementaria	24
Sesión 5: Comenzar Historias y Personajes Únicos.....	27
Hoja del Estudiante Actividad 5: Modo y Tono.....	28
Sesión 6: Estrategia, Modo y Tono	29
Hoja del Estudiante Actividad 6: <i>Creatables</i> (seres que se pueden crear)	32
Sesión 7: Cambiar Comportamientos Usar Páginas, Establecer, y Cambiar Perspectivas	33
Hoja del Estudiante Actividad 7: Ángulos de Cámara y Cambiar Comportamiento	37
Sesión 8: <i>Power Ups</i> (aumento de poder), <i>Health</i> (Salud), y <i>Timer</i> (Cronómetro).....	38
Hoja del Estudiante Actividad 8: Cronómetros, Salud y “ <i>Power Ups</i> ”.....	43
Sesión 9: Más sobre Puntaje – Fundamentos para la Comunicación	44
Hoja del Estudiante Actividad 9: Puntaje y Comportamiento	45

Final de Kodu: Presentaciones de los Estudiantes	46
Presentaciones de los estudiantes	46

Currículo de Kodu: Teclado y Ratón

¡Bienvenidos a la versión para teclado del Currículo de Kodu (versión teclado)! Lo que sigue es un recurso para usar **Kodu** como parte de una clase, programa para después de la escuela, o campo de verano. Esto proporciona un alcance y secuencia básicos que tiene el propósito de introducir instructores y estudiantes a la aplicación de **Kodu**, y conceptos básicos de programación de computadores, y sugiere formas en que Kodu podría ser usado para complementar el currículo tradicional.

La meta principal de **Kodu** es exponer a los estudiantes a la programación de computadores en un entorno divertido, mientras se les hace emocionar acerca de carreras en la ciencia de la computación permitiéndoles crear sus propios juegos. Esto los hace productores de sus propios medios en vez de solamente consumidores de ellos. Más específicamente, inmediatamente de usar **Kodu**, los estudiantes:

- ✓ Entenderán mejor los pasos involucrados en la creación de un programa de computador
- ✓ Mejorarán las habilidades de resolución de problemas, y fomentará las prácticas de solución de problemas
- ✓ Seguirán direcciones en línea y fuera de línea en forma más fluida
- ✓ Aprenderán a componer historias en un formato alternativo y a través de medios variables
- ✓ Practicarán implícitamente matemáticas a través de bonificaciones y puntajes
- ✓ Desarrollarán actitudes más positivas hacia la programación de computadores
- ✓ Crearán juegos cada vez más complejos mostrando así un entendimiento más profundo para secuencias de codificación complejas
- ✓ Mostrarán pruebas de toma de perspectiva y empatía en el juego
- ✓ Trabajarán colaborativamente para crear soluciones innovadoras.

Descripción de Kodu

La interfaz de usuario es el cimiento para trabajar con Kodu. El lenguaje es sencillo y completamente basado en íconos. Los programas consisten en páginas, las cuales se desglosan en reglas, las cuales son adicionalmente divididas en condiciones y acciones. El lenguaje **Kodu** está diseñado específicamente para desarrollo de juegos y proporciona conceptos básicos especializados (los sustantivos, adjetivos, y verbos del lenguaje) derivados de escenarios de juegos. Los programas están expresados en términos físicos, usando conceptos tales como visión, oído y tiempo para controlar el comportamiento del personaje. **Kodu** puede expresar conceptos avanzados de diseño de juegos en una forma sencilla, directa e intuitiva.

Enseñando con Kodu

Sugerimos que la mejor manera de enseñar **Kodu** es a través de pedagogía de estudio con mini-lecciones intercaladas en todas partes como un medio para mover a los estudiantes en su descubrimiento de las herramientas mientras se ejecutan prácticas de solución de problemas colaborativas. Sin embargo, nótese que el entorno donde **Kodu** es presentado, junto con las inclinaciones pedagógicas de los facilitadores, podría requerir una estructura más suelta o más ajustada. En pocas palabras, el instructor debe tomar aquella pista que él o ella sienta le queda mejor a la clase y a su estilo de enseñanza. Nuestra meta es ayudar a tratar las muchas formas en que los estudiantes responden a la tecnología—a algunos les gusta ser dirigidos en su avance y otros tienden a avanzar con ímpetu, experimentando a través de prueba y error. En este currículo hay campo para ambos tipos de estudiantes.

Acerca de los Desarrolladores

Kodu fue desarrollado por un equipo de programadores de investigación en Microsoft quienes son apasionados por que los niños se diviertan y sean retados a medida que aprenden cómo programar.

Para Mayor Información

Visite <http://research.microsoft.com/en-us/projects/kodu/> para mayor información acerca de **Kodu** y sus desarrolladores.

Visite el blog de **Kodu** para ver lo que dicen otros sobre **Kodu** o publique sus ideas acerca del programa.

Hay algunas rutinas y prácticas pedagógicas que nosotros sugerimos para ayudar a facilitar la clase y enriquecer la experiencia de aprendizaje de los estudiantes. Primero, aunque es importante una instrucción explícita acerca de las herramientas y los criterios para los proyectos finales, usted deberá dejar que los estudiantes experimenten alegremente con las herramientas. Esto significa que ellos cometerán algunos errores al tratar de resolverlo.

Cultura de Solución de Problemas

Para ayudar a facilitar la comisión de errores deliberada, a los estudiantes se les debe recordar continuamente que ellos, como un colectivo, tienen las claves para la solución de problemas. “**Three-before-me**” (tres antes que yo) es una técnica usada para ayudar a estudiantes en conversaciones colaborativas de solución de problemas—en palabras simples, si los estudiantes no pueden resolver un asunto, ellos le preguntan a otros tres estudiantes antes de buscar al instructor. Otro método es escribir problemas en un “**post-it**” y colocarlos en un espacio colectivo donde los estudiantes comparten y se responden entre ellos; muy parecidos a un tablero digital de discusión. A medida que los estudiantes responden preguntas con otro “**post-it**” electrónico, las preguntas se mueven al lado de “resueltas” del tablero.

Una Cultura de Crítica

La pedagogía del estudio permite un espacio para que los estudiantes discutan su trabajo y reciban una retroalimentación—este procedimiento está integrado durante el momento de emprender la creación del juego, es muy importante plasmar todas las ideas y el texto en un borrador, como lo es tradicional de la escritura, o al final de la creación del producto. Hay muchas formas de establecer esta cultura a la clase. Un método es hacer que los estudiantes

trabajen en grupos pequeños, tomando turnos para discutir las intenciones, retos y éxitos de cada proyecto mientras otros en el grupo dan retroalimentación cálida y constructiva. Con frecuencia, esto necesita ser moldeado. Otra estrategia de implementación podría incluir exhibiciones regulares de toda la clase durante las cuales los estudiantes investigan todos los juegos en varias etapas en desarrollo y hacen preguntas de los proyectos de otros estudiantes. De nuevo, esta estrategia probablemente necesitará ser perfeccionada, quizás a través de una metodología de pecera.

Sugerimos que usted use el formato de estudio al menos una vez a la semana si ustedes se reúnen diariamente. Asimismo, le sugerimos que la clase sea cerrada con una discusión que esboce los éxitos y desafíos de la sesión de ese día. Esto puede estar directamente relacionado con asuntos de **Kodu** o con procesos internos de clase más generales y asuntos del clima del salón de clases.

Práctica Reflexiva

También se sugiere que usted y sus estudiantes mantengan un diario acerca de su trabajo con **Kodu**. Es mejor si la escritura del diario se convierte en parte del ritual de la clase, quizás al final de cada sesión. Preguntas orientadas ayudarían a facilitar este proceso. Estos mismos diarios igualmente, podrían albergar la planeación de juegos por parte de estudiantes una vez que ellos comiencen a trabajar en sus proyectos de juegos.

Exhibición del trabajo de los estudiantes

Nosotros aconsejamos que usted termine su unidad de **Kodu** con una exhibición del trabajo de los estudiantes. Los estudiantes pueden presentar sus juegos a una audiencia de pares, padres y otros adultos. Para que tenga mayor interés, considere dar los componentes de un juego exitoso con los estudiantes. Esto puede ser usado como los criterios para evaluar los juegos

Trabajando con el Teclado y el Ratón

Kodu fue diseñado originalmente con el usuario jugando y editando mundos usando un controlador de **Xbox**. Teniendo claridad de que los controladores pueden tener costos prohibitivos para muchas escuelas, centros comunitarios y hogares, los desarrolladores de **Kodu** crearon la versión para teclado del software. Sin embargo, la mayoría de los mundos del juego dentro de **Kodu** todavía requieren el controlador para jugarlos, dado que aún no han sido convertidos. Esto puede presentar algunos asuntos cuando los usuarios comienzan a trabajar con el programa. Al mismo tiempo, siempre fuerza a los usuarios a hacer que los mundos de los juegos funcionen para que ellos puedan editar los códigos.

Hay dos métodos principales para hacer que un **avatar** o personajes controlados por el jugador se muevan en la versión para teclado.

El **Método 1** usa una dirección más generalizada a través de las teclas de flecha. En la estructura del código, simplemente escriba en el área de código **WHEN: keyboard DO: Move**. La flecha hacia arriba representa el movimiento hacia adelante, y las flechas derecha e izquierda representan giro hacia la derecha y hacia la izquierda respectivamente. Entonces, este método no confía en la brújula norte, sur, este y oeste.

El **Método 2** es por medio de las teclas de flechas siendo programadas a una dirección particular (Norte/Arriba, Sur/Abajo, Este/Izquierda y Oeste/Derecha). Sin embargo, usar este método puede presentar retos ya que el Norte podría no ser arriba (o el Sur podría no ser abajo, etc.) dependiendo de cómo haya sido creado el paisaje en relación con dónde se colocan los jugadores o personajes. Esto puede ser confuso cuando se está jugando un juego, pero de nuevo, los mundos y los personajes pueden ser reprogramados y rediseñados para acomodar un teclado y un ratón. Cuando se use este método, el indicador de la brújula es una herramienta esencial cuando se naveguen mundos usando las teclas de flechas. Algunos mundos no tienen la brújula en la interfaz que indica la dirección; para mostrar la brújula, ingrese al modo de Edición dentro de un mundo de juego presionando la tecla **“Escape”** y seleccionando el ícono final en la lista (representado como una llave de tuercas y paisaje de montaña). Luego, flecha hacia abajo para **“Show Compass”** (Mostrar Brújula) y asegurarse de que sea seleccionado.



Alcance y Secuencia del Currículo de Kodu

Sesión 1	Navegar, Introducción a conceptos de programación, agregar objetos
Sesión 2	Crear un Paisaje
Sesión 3	Usar el Controlador para mover personajes, Crear Rutas, y Establecer comportamientos
Sesión 4	Hacer clones y “Creatables” (seres que pueden ser creados)
Sesión 5	Iniciar historias y personajes únicos
Sesión 6	Estrategia, Modo, y Tono
Sesión 7	Cambiar comportamientos usando páginas, Establecer y cambiar perspectivas
Sesión 8	“Power Ups” (aumentos de poder), Salud, “Timer” (Cronómetro)
Sesión 9	Más sobre Puntajes—Elementos básicos para la Comunicación

Sesión 1: Navegar, Introducción a Conceptos de programación, Adicionar objetos

Cuando terminen esta sesión, los estudiantes podrán:

- Navegar el macro entorno de **Kodu** usando un teclado y un ratón
- Entender los principios fundamentales de la programación
- Tener acceso al modo de programación de **Kodu**, potencialmente ajustando código sencillo para un propósito específico

Encuesta—Ilegar a conocerse cada uno a través de la encuesta

Antes que se vayan a aventurar en **Kodu**, por favor haga que sus estudiantes tomen la encuesta de **Kodu**. La encuesta actuará como un trampolín para que exista una conversación que le permita a usted llegar a conocer mejor a sus estudiantes. Los datos recopilados en la encuesta ayudan a los desarrolladores de **Kodu**, y todavía proporciona a los profesores información útil acerca de los tipos de prácticas de tecnología en las cuales se comprometen sus muchachos.

Ejercitarse en Programar un “Avatar” ingenuo

Un ejercicio en clase es útil para aleccionar a los estudiantes con alguna idea de lo que es programar juegos en **Kodu**.

Materiales:

- 3 manzanas rojas (bolas o algún otro objeto de color funcionará)
- 2 manzanas verdes
- 1 bolsa
- 1 o 2 vendas para los ojos

Informar a la clase que en Kodu ellos necesitarán:

- Seleccionar un personaje y hacerlo que se comporte y reaccione de ciertas maneras,
- Construir un entorno en el cual operen los personajes
- Llenar ese entorno con objetos con los cuales los personajes interactúan o no
- Crear reglas y reacciones que permitan a los personajes funcionar en el entorno

Solicite a un estudiante voluntario que actúe como un personaje de **Kodu**. Hágalo(a) sentarse en un asiento en la mitad del salón con los ojos vendados. Luego pregunte a los otros estudiantes cómo debe ser arreglado el salón para crear un entorno de **Kodu**. Dígales que usted tiene cinco (5) objetos para colocar en el entorno (3 manzanas rojas y 2 verdes). Colóquelas en diferentes puntos del salón. Infórmeles que necesitamos dirigir nuestro

personaje de **Kodu** para que encuentre y levante manzanas. Note que cada acción tiene que ser esbozada y descrita en detalle.

Para ayudarles a organizar sus “**comandos**” informe a los estudiantes que el personaje solamente le escucha a usted, entonces ellos necesitan filtrar los comandos. Pida aclaración y detalles adicionales si algo es vago o difícil de ejecutar. Dependiendo de qué tan enérgicos sean sus estudiantes, usted podrá necesitar apelar a los estudiantes para que ayuden con el manejo del salón de clases.

Continúe haciendo preguntas acerca de ¿Cómo, Cuándo, y Dónde?. Siéntase libre de quitarle la venda al estudiante si se le sugiere que vea (este tipo de instrucción se le conoce como un “comando” en **Kodu**).

Durante el ejercicio considere las siguientes opciones a las cuales Kodu puede responder:

- Adelante y atrás
- Norte, Sur, Este, Oeste
- Rápido y despacio
- Saltar y voltear
- Golpearse contra cosas
- Puede ser programado para moverse hacia y lejos de ciertos objetos
- **Kodu** puede ver, oír, y distinguir colores
- **Kodu** puede expresar amor, ira, tristeza, y locura
- **Kodu** también puede decir cosas a través de texto
- Inclusive, **Kodu** puede esperar en lugares
- **Kodu** puede moverse entre lugares dentro de una cierta cantidad de tiempo

Cosas para recordar: Todas estas acciones y reacciones necesitan ser programadas por el usuario. Entonces a medida que usted toma comandos de la clase, asegúrese de preguntarles continuamente ¿Cómo, Cuándo, Por qué, y Dónde? para tener cada vez más detalle acerca de la acción.

Un escenario típico podría ser:

Profesor: Queremos que **Kodu** haga algo que involucre estas manzanas. Tratemos sólo de mover nuestro **Kodu** a la manzana y ponerla en su bolsa. ¿Qué sería la primera cosa que le diríamos que haga?

Estudiante: ¿Muévase a la manzana?

Profesor: ¿Cómo? ¿Él Camina?

Estudiante: Sí, él camina

Profesor: ¿Rápido o lento?

Estudiante: Rápido.

Profesor: ¿Cómo sabe cuándo? ¿Qué lo impulsa a moverse?

A medida que transpira el escenario, podría ser útil escribir los **comandos** en el tablero para que el curso pueda recordarlos en forma fácil. Una vez se haya establecido un conjunto coherente de comandos, ponga su **Kodu** en movimiento.

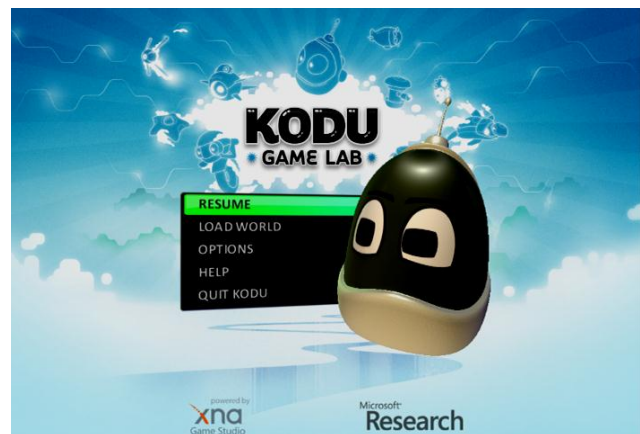
Presentando la Navegación y la Página Principal de Kodu

Antes de hacer que los estudiantes ingresen a **Kodu**, muéstreles la estructura básica del programa a través de la **Página Principal**. Ella tiene cinco opciones que el usuario o bien va pasando con las flechas o selecciona haciendo clic con el botón izquierdo del ratón o presionando “**Enter**”.

1. “Resume” (Reanudar)

Abre cualquier juego o nivel que fuera editado de último por el usuario. Si el usuario abre esta ventana, aparecerá el último juego que fue programado o abierto.

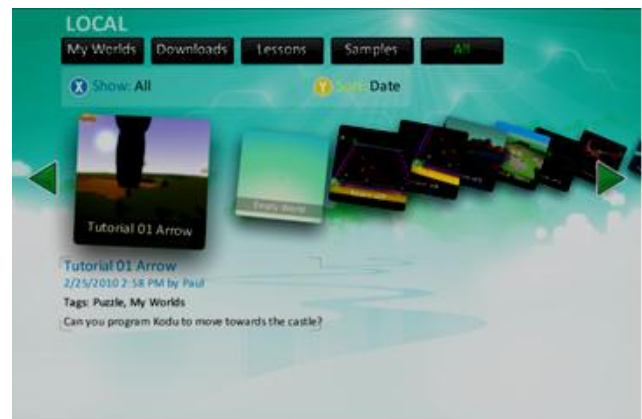
Para salir de esta ventana, haga clic en el ícono “**House**” y luego seleccione “**Exit to Main Menu**” (Salir a Menú principal).



2. Load World” (Cargar Mundo)

Abre una serie de juegos o niveles de los cuales el usuario puede seleccionar. Los niveles pueden ser examinados usando ya sean las teclas de flechas o haciendo clic con el ratón sobre las flechas que aparecen en la pantalla.

Para regresar al **Main Menu**, los usuarios presionan **Escape**.



3. “Options” (Opciones)

Le proporciona al usuario la habilidad de cambiar algunas características amplias del nivel del juego. Para navegar por esta lista, los usuarios o bien usan las teclas de flechas o posan el cursor del ratón sobre las flechas verdes sobre la pantalla. **Escape** regresará al usuario al **Main Menu**.

4. “*Help*” (Ayuda)

Ofrece una explicación rápida de **Kodu**. Los usuarios presionan *Escape* para salir de este menú.

5. “*Quit*” (Salir)

Permite que los usuarios salgan del programa.

Introducción a Comer Manzanas

Para hacer eco a la actividad de mover y codificar figuras en el salón, pasar al nivel “**Activity 1 Eating Apples**” (Actividad 1 Comer Manzanas) en **Kodu**. Navegue al nivel durante la clase, y diga a los estudiantes que tomen nota de todo lo que está sucediendo en la pantalla.

Seleccione “**play**” para correr el programa, y pregunte qué notaron ellos.

- **Kodu** se come las manzanas.
- **Kodu** se mueve de manzana a manzana.
- **Kodu** no se come la manzana verde.
- El árbol produce más manzanas y ellas se apilan detrás de **Kodu**.

Ahora, muestre a los estudiantes el código de programa que está corriendo este conjunto de comportamientos—**Kodu** y el árbol. **El código** puede ser leído presionando la tecla “**Esc**” cuando el juego es iniciado.

A continuación, seleccione el ícono de **Kodu** de la barra de herramientas y dé vueltas al cursor sobre el objeto (**Kodu** o el árbol) del cual usted quiere leer el código. El objeto se iluminará. Hacer clic en el botón derecho mientras se mantiene presionado permitirá al usuario mover el objeto, mientras que hacer clic en el botón izquierdo traerá un montón de opciones para que el usuario escoja.

“**Program**” (**Programa**) hará que aparezca el menú de código. Lean el código juntos y pregunte si ellos ven estos comportamientos en el código tanto para el **Kodu** como para el árbol. Después de leer el código, edite el mundo mientras les pregunta a los estudiantes cómo puede cambiar el código para el conjunto de comportamientos.

- Agregue una manzana. [Esto se produce seleccionando a **Kodu** de la barra de herramientas con el ratón, y luego haciendo clic en el paisaje donde usted quiere que vaya la manzana. Aparecerá el anillo de selección de objetos, y seleccionará el mosaico de la manzana.]
- Haga que una manzana en la secuencia se vuelva verde; asegúrese que **Kodu** no se la coma. [Haga clic sobre o pase el cursor sobre una de las manzanas en el círculo de manzanas y aparecerá la paleta de colores en la parte superior. Usando las teclas de flechas, haga que una de las manzanas se vuelva verde.]
- Como un reto, ajuste el comportamiento de **Kodu** de forma que dé la vuelta para buscar más manzanas cuando no vea ninguna. [Hay varios niveles para este comando dependiendo de qué tan lejos usted quiere programar a **Kodu**. La secuencia siguiente debe permitir que **Kodu** busque manzanas en forma continua. Simplemente agregue el código para dar vuelta en la parte inferior de la secuencia de programación no funcionará. Siga esta secuencia para hacer que **Kodu** dé vuelta y busque más manzanas.

(Línea 1) **When:** see none apples Do: move turn quickly.

(Línea 2) **When:** see apples red Do: move toward.

(Línea 3) *When: bump, apples red Do: eat.*

(Línea 4) *When: see none apples red Do: move turn quickly*

(Línea 5) *When: see apples red, Do: move toward quickly.*

Esta actividad es muy meritoria para los estudiantes ya que modela la prueba y error que con frecuencia está involucrada con la programación. Posteriormente de modelar las estrategias, buscar el código, y ensamblar el proceso, haga que los estudiantes practiquen con “**Activity 1 Eating Apples**” (Actividad 1 Comer Manzanas).

Reflexión

Después que los estudiantes trabajen toda la **Activity 1 Eating Apples** (Actividad 1 Comer Manzanas), diga a los muchachos que salven sus proyectos y apaguen sus monitores. Es el momento de hacer reflexión. Haga a los estudiantes las siguientes preguntas:

- ¿Qué fue desafiante?
- ¿Qué fue fácil?
- ¿Qué fue un éxito?
- ¿Qué aprendieron ellos o qué sabían ellos ya?
- ¿Qué quieren aprender a continuación?

Lo que usted puede esperar de la Actividad de Estudiantes

El tiempo libre es esencial para que los estudiantes se familiaricen con la herramienta y algunos conceptos generales de programación. Usted encontrará que la mayoría de los estudiantes trabajarán a través del listado que usted proporciona y luego explorarán las varias características por ellos mismos—adicionando más personajes y objetos y dándoles comportamientos.

Algunos estudiantes inclusive trabajarán con el paisaje. Otros comenzarán a jugar juegos y ni siquiera mirarán la lista de cosas por hacer. Lo que usted permita depende de usted, pero siempre es bueno reagruparse para compartir sus experiencias y pensamientos.

Incluso, si algunos estudiantes están trabajando en herramientas que usted estará cubriendo en las próximas lecciones, entonces considere hacer que esos estudiantes enseñen a otros discípulos cómo usar esa característica.



Hoja del Estudiante Actividad 1: Comer Manzanas

Objetivos: Adicionar el elemento, cambiar color, seleccionar objeto, crear texto con base en el evento, cambiar movimiento basado en no-evento

Instrucciones: Acabamos de pasar por algunos de estos pasos como grupo en la **Activity 1 Eating Apples**. Ahora es su turno de probarla. A medida que diligencie cada una de las siguientes preguntas, TÁCHELA DE SU LISTA.

Una vez haya terminado, revise con su asesor y haga que mire lo que usted ha terminado. Las palabras subrayadas tienen el propósito de ser pistas para terminar su código.

Lista de Verificación de cosas por hacer

Adicione una manzana a la secuencia de manzanas—hágala de color azul.

Seleccione una manzana existente y cambie su color a azul.

Haga que **Kodu diga** (se encuentra en acciones) “tengo hambre” cuando se choque con la manzana azul.

Si se están apilando manzanas bajo el árbol inmediatamente de que la secuencia de **Kodu** esté hecha, haga que **Kodu dé la vuelta** completa de forma tal que pueda ver las manzanas y se las coma.

Si las manzanas no se están apilando, haga una manzana detrás del árbol al cual es atraído el **Kodu**. Esto debe hacer que las manzanas comiencen a apilarse. Cuando usted lo haya programado entonces las manzanas se están apilando, entonces codifique a **Kodu** para que dé la vuelta completa cuando él no vea ninguna y busque las otras manzanas.

Una vez que usted complete la lista de cosas por hacer, ensaye adicionando otros objetos, ajustando los comportamientos de **Kodu**, y cambiando los entornos. Por favor salve su trabajo. Pida ayuda si la necesita.

Actividad de Desafío

Como una actividad de desafío, vaya al **Tutorial 01 v2** y haga lo que pida el **Kodu**. Asimismo, vea si usted puede reprogramar el castillo para que se comporte en una forma diferente una vez que es chocado, y trate de averiguar cómo puede la cámara seguir a **Kodu** en su arduo viaje al castillo. Otro escenario de comida de manzana es presentado a usted en la “**Technique: Eating only certain apples v07**” (Técnica: Comer sólo ciertas manzanas v07). Dele un vistazo y vea si puede reprogramar de acuerdo con la pantalla de entrada.

Sesión 2: Crear un Paisaje

Cuando se termine esta sesión, los estudiantes podrán:

- Cambiar y crear un entorno de **Kodu**
- Usar mosaicos como decoración y el desarrollo de tono y modo del mundo de juegos

Crear y cambiar el paisaje de **Kodu** es una de las actividades iniciales más interesantes para los usuarios iniciales de **Kodu**. Con frecuencia ellos gastan horas cambiando y diseñando paisajes intrincados en los cuales operan sus juegos y animaciones. Los paisajes establecen el tono para las acciones que siguen. Por estas razones, nosotros dedicamos una buena cantidad de tiempo a las características del paisaje y la experimentación de los estudiantes. Las intenciones de las actividades son que los niños comiencen a construir mundos desde el principio (literalmente) y pensando en forma creativa acerca de su uso de mosaicos y cómo ensamblarlos para un efecto particular.

Instrucción para toda la clase

Utilizando un computador conectado a un proyector, solicite estudiantes voluntarios para que operen varias de las características del paisaje—algunos de ellos probablemente habrán usado la característica del paisaje durante la sesión previa. Abriendo un mundo vacío pida a los estudiantes que demuestren cómo adicionar y eliminar tierra, así como darle color y textura. Pídeles verbalizar lo que ellos están haciendo para que todo el curso pueda oír su proceso de pensamiento e instrucciones. A medida que usted se mueva a hablar acerca de modo y tono, asegúrese de tener estudiantes tratando de definir estos términos. Inclusive estudiantes en escuela media podrían tener problema con lo que significan tono y modo, entonces sea explícito en sus instrucciones y en forma repetida vuelva a ellos a medida que la clase progrese durante todas las lecciones.

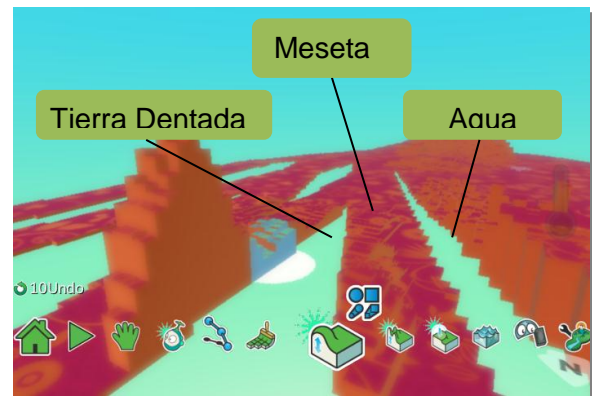
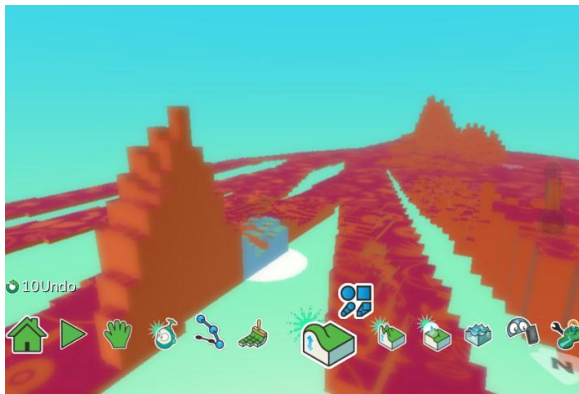
- Adicionar/eliminar tierra

Escoja una brocha verde en la barra de herramientas. Seleccione el material de paisaje apropiado haciendo clic en el ícono de materiales que está en la parte superior izquierda de la brocha de pintar. Luego, use las flechas para hacer una selección. Una vez se haya encontrado una que parezca apropiada, haga clic sobre ella. Para aplicar tierra, presione y mantenga presionado el botón izquierdo del ratón mientras mueva el ratón para “**paint**” (pintar) el paisaje. Presionando



hacia abajo el botón derecho del ratón, y moviendo el ratón, el usuario puede borrar formaciones de tierra.

- **Escoger un borde apropiado**—Escoja la brocha verde en la barra de herramientas y la brocha apropiada haciendo clic en el ícono de formas geométricas en la parte superior derecha del ícono. Vaya pasando la flecha por las alternativas y seleccione el borde deseado. La brocha cuadrada dará a los usuarios un borde plano mientras que la circular proporcionará un borde más redondeado. Los íconos ampliados cuadrado y circular permiten a los usuarios dibujar tramos de tierra con un borde más claro, y el ícono de la varita mágica proporciona una forma fácil de cambiar el color y textura de un paisaje en masa con facilidad.
- **Crear colinas y valles; usar la característica de suavizar**—Una vez que es creado un paisaje, la tierra puede ser elevada y bajada. Hay varios efectos de los cuales escoger cuando se esté bajando y elevando tierra. El primer ícono en las series mostradas a continuación permite que la tierra sea elevada o bajada como un colectivo basado en dónde se coloque la brocha. El segundo ícono en las series eleva y baja más dentadamente, y el tercer ícono permite que el usuario aplane una superficie, así creando una meseta.



- **Crear una isla o un lago**—El ícono de agua en la barra de herramientas permite a los usuarios adicionar lagos, mares y océanos a sus mundos. Para adicionar estos entornos, el diseñador del mundo tiene que crear primero una extensión de tierra sobre la cual descansa el agua. Note en los pantallazos que aparecen arriba, cómo el agua no llenó los espacios en los cuales no se dibujó tierra. Cuando se hace clic en el ícono del agua, a los usuarios se les dan opciones para el color del agua. Esto puede ser examinado usando las flechas y el ratón.
- **Ajustar la decoración para agua y cielo**—el icono de “**settings**” (configuración) al final de la barra de herramientas proporciona algunas de las características más poderosas en **Kodu** y tienen impacto tanto en el movimiento en, y apariencia del mundo. Por tanto, hay algunas decoraciones que influyen una gran cantidad en el tono y modo. Aquí hay algunas decoraciones que tienen un impacto sobre modo y tono, así como la



creación general del mundo.

- **“Glass Walls”** (Paredes de Vidrio) es la decoración por omisión para todos los mundos de juego y actúa para contener el juego dentro del entorno dibujado. Sin las paredes, los personajes pueden inclinarse dentro del abismo por fuera del mundo de juegos.
- **“Show Compass”** (Mostrar Brújula) permite al usuario entender a qué dirección se está moviendo. Esto es particularmente importante cuando los usuarios comienzan a mover personajes y a usar teclas de flechas para impulsar el movimiento.
- **“Wave Height”** (Altura de Olas) ajusta el tamaño de las olas en el mundo de juegos
- **“Water Strength”** (Fuerza del Agua) cambia el nivel de distorsión del agua.
- **“Sky”** (Cielo) proporciona al usuario tipos atmosféricos variables.
- **“Lighting”** (Iluminación) establece la cantidad y tipo de luz que es vertida en su mundo.

A continuación de que usted haya cubierto cada una de estas áreas con sus estudiantes actuando como impulsores en la creación del mundo de juegos, haga que ellos experimenten con las herramientas que hacen un paisaje, así como crear tono y modo. El contraste en los proyectos de los estudiantes proporcionará material para un diálogo interesante acerca de la decoración/ disposición y su rol en un juego.

Cambiar la Línea de Visión

Mientras los estudiantes están construyendo sus mundos, podrían encontrar que es difícil ver lo que en realidad están creando dependiendo del ángulo en el cual ven sus mundos. Para ayudar a navegar sus mundos a medida que ellos los estén construyendo, muéstreles la **“Green Hand”** (Mano Verde) en la barra de herramientas. Cuando se presiona el botón izquierdo del ratón, el cursor permite al usuario empujar el paisaje hacia adelante, hacia atrás, y a los lados. Adicionalmente, el botón central del ratón para desplazamiento permite al usuario hacer acercamientos y alejamientos, y si el usuario mantiene presionado los botones izquierdo y derecho del ratón al mismo tiempo, una línea de visión del mundo de juegos puede ser trasladada vertical y horizontalmente.



Reflexión

Durante los últimos 20 minutos de clase, haga que los estudiantes compartan sus mundos. Pídeles compartir entre ellos lo que encuentren interesante cerca de los mundos de los demás. Si ellos terminan esta actividad temprano, diga a los estudiantes que se sientan libres para explorar los otros mundos que ya existen en **Kodu**.

Lo que usted puede esperar de la Actividad de Estudiantes

Las herramientas de paisaje son particularmente cautivadoras para los estudiantes. Algunos estudiantes gastarán horas construyendo y revisando sus entornos para crear un efecto específico. De hecho, algunos de los muchachos se volverán tan enfrascados en su mundo, que ellos colocarán sus esfuerzos en el paisaje en contraposición al desarrollo de personajes, estrategia de juego, argumentos y usabilidad. Es importante recordar que la gente tiende a componer y crear en formas diferentes. Considere escribir; algunas veces la intención del autor es la satisfacción personal por encima del interés de la audiencia. Esto de la misma manera, podría ser cierto para los estudiantes mientras estén trabajando en **Kodu**. Si usted encuentra que este es el caso, sería beneficioso discutir estos enfoques a la composición e intentar como clase una vez que haya ingresado a las discusiones de caracterización y argumento.

Enlace a Campo de Actividad — Geografía

Si usted piensa que es apropiado, considere hacer un enlace de enseñanza a geografía. Se pueden diseñar y demostrar paisajes increíbles con **Kodu**.

Dependiendo del grado del estudiante, usted podría pedirle que ilustre diferentes términos geográficos: colinas, valles, montañas, península, peñasco, isla, bahía, istmo, volcán, meseta, canal, etc.

Kodu puede también ilustrar cambios a paisajes: considere por ejemplo, ilustrar erosión, cambios a la extensión de tierra debido a terremotos y volcanes, actividad de ventisca, etc.

Un ejercicio interesante para estudiantes podría ser crear un mapa topográfico de una montaña, río, estado, provincia, o país, quizás aún ilustrando la misma masa de tierra antes y detrás de un evento geológico.



Hoja de Estudiantes - Actividad 2: Hacer Paisajes

Objetivos: Crear tierra con textura, adicionar agua, árboles, rocas, etc.

Direcciones: Nosotros acabamos de revisar las diferentes herramientas para crear un paisaje. Ahora es su turno de intentar hacerlo. A medida que usted complete cada una de las siguientes cosas, márquela en su lista como realizada.

Verifique con su instructor una vez que haya terminado.

Listas de Verificación de cosas por hacer:

Ir a “**Empty World**” (Mundo Vacío).

Crear una extensión de tierra con:

- Al menos dos tipos de materiales
- Crear colinas onduladas, montañas (con un pico blanco), y valles

Hacer una isla o dos fuera de la costa de su tierra

Adicionar agua ya sea como un río, lago, océano o todos los anteriores

Crear un bosque mágico en algún lugar en su paisaje (Usted puede definir mágico en cualquier forma que quiera). Hay varios objetos de donde escoger: árboles, rocas, estrellas, monedas, etc.)

Crear nubes de tormenta sobre una parte de su paisaje.

Nivel de Desafío

Más tarde de que usted haya creado su mundo, vea si puede encontrar dónde cambiar el modo y tono del juego. Específicamente, trate cambiando algunas de las configuraciones. Investigue las siguientes disposiciones y note cómo cambian ellos el significado de su mundo:

Altura de la nube

Fuerza del agua

Cielo

Iluminación

Brisa

¿Cómo pueden los ajustes a su configuración cambiar la sensación de su mundo? Venga preparado para compartir su mundo con sus compañeros de clase.

Sesión 3: Usar el Controlador para Mover Personajes, Crear Rutas, y Establecer Comportamientos

Cuando haya terminado, los estudiantes podrán:

- Usar el teclado para mover personajes en un mundo de juegos
- Crear rutas en las cuales se moverán los personajes
- Dar comportamientos a objetos

Hasta ahora en los planes de la lección, a los estudiantes no se les ha pedido programar personajes para hacerlos mover (excepto lo que está en el “**Challenge Level**” (Nivel de Desafío) en la Sesión 1 usando **Tutorial 01 v3**). Esto no quiere decir que algunos estudiantes ya no se hayan dado cuenta de cómo hacer esto. Con toda probabilidad algunos lo han hecho, mientras que otros han estado contentos de seguir el libreto de las actividades del estudiante. En cualquier caso, podría ser productivo hacer que los estudiantes otra vez se lleven entre ellos en estas actividades. Si hay estudiantes que hayan trabajado en forma avanzada, haga que ellos demuestren a los otros cómo programar el movimiento guiado por el usuario, crear rutas, y dar comportamientos a objetos.

Instrucción para Todo el Grupo

Para los estudiantes que hicieron el **Tutorial 01 v3**, este sería un buen momento para que ellos compartan la forma en que resolvieron el rompecabezas del juego. Usando el proyector, pida a



un estudiante voluntario que demuestre cómo hizo que **Kodu** fuera al castillo. Como un recordatorio, inmediatamente de abrir el mundo de juego de **Tutorial 01 v3**, presione “**Esc**” y haga clic sobre el ícono de **Kodu** en la barra de herramientas. Coloque el cursor sobre el **Avatar de Kodu** y presione el botón derecho del ratón. Seleccione “**Program**” (Programa). El código actual está debajo. Hay dos formas principales para hacer que **Kodu** se mueva al castillo. En cualquiera de los casos, el usuario tiene que entrar en modo de edición, seleccionar el ícono de **Kodu**, y editar su comportamiento. El **Kodu** puede ser programado ya sea para moverse como si fuera mediante animación o mediante control del usuario vía las teclas de flechas.

Para animar a **Kodu** a que se mueva al castillo, adicione el siguiente código a la secuencia.



Para controlar los movimientos de **Kodu**, se pueden usar dos secuencias de programación. La primera usa direcciones generalizadas de **adelante**, **derecha**, e **izquierda**.



Un **segundo método** para mover a **Kodu** incluye usar las direcciones Norte, Sur, Este y Oeste.



Si el estudiante demuestra uno de los métodos anteriores, asegúrese de sugerir un método alternativo para mostrar que el problema puede ser resuelto de múltiples maneras. Asimismo,

asegúrese de hacer que los estudiantes lean el código que precede la nueva línea que se ingresa. Es bueno para los estudiantes practicar leyendo la sintaxis del código.

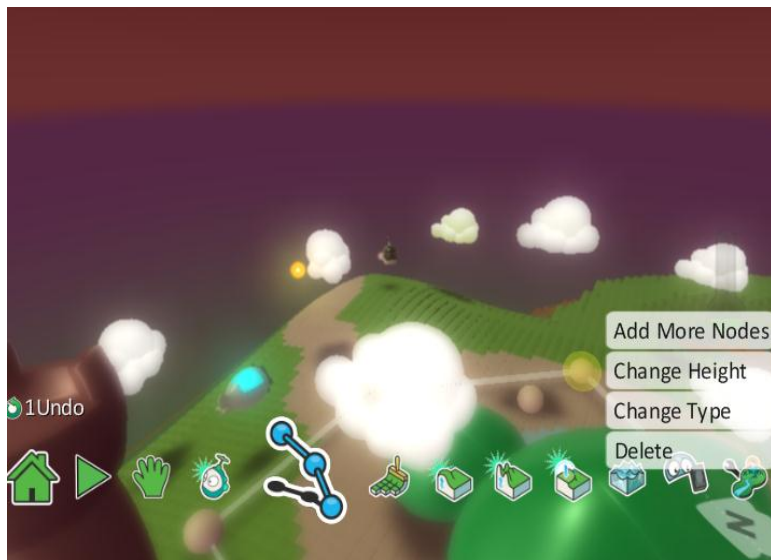
Ahora vaya a **Idyll KB**. Este juego tiene un número de componentes que vienen de la lección. En vez de jugar el juego inmediatamente, haga que los estudiantes lean el código ya sea en grupos pequeños o en forma colectiva. Si ellos trabajan en grupos pequeños y reportan al grupo más grande, sea específico acerca del código que usted quiere que miren en: **Kodu**, los castillos, el dirigible, y el árbol.

Si está trabajando con toda la clase, inicie con el **Kodu**—el código es bastante directo. Aunque usted puede ir a la Página 2 y leer el código, las páginas serán cubiertas en la siguiente lección. Ahora, haga correr el juego. Pregunte a los estudiantes qué está pasando en el juego que ellos no vieron en el código. Probablemente ellos notarán que los castillos desaparecen y crean espirales. Ellos todavía notarán el dirigible moviéndose. Y, ellos aún pueden notar que se están acumulando puntos—esto es programado a través del árbol. Regrese y dé una mirada al código para cada uno de estos componentes. Lo que es de más interés para nosotros en esta lección son el castillo y el dirigible. El puntaje será cubierto en la Lección 6. Un espiral de la misma manera, está codificado dentro del juego como un “*creatable*” (ser que puede ser creado), pero esta característica será cubierta en la Lección 4. Lea el código para el castillo. De nuevo, es directo—cuando **Kodu** choca con el castillo este libera un espiral y explota. Luego, lea el código del dirigible—sólo hay una frase de **DO, ningún WHEN**. Pregunte a los estudiantes qué significa esto.

“**Paths**” (Rutas)

La idea de una ruta será probablemente nueva para los estudiantes. Muéstreles que la altura de la ruta puede ser elevada o bajada resaltando un nodo, presionando el botón izquierdo del ratón, y luego escogiendo “**Change Height**” (Cambiar Altura). También note que el color y tipo de la ruta necesitan ser especificados para que el dirigible se mueva por la ruta.

Mientras esté mirando a **Idyll KB**, pregúnteles cómo crearían ellos una segunda ruta para un globo de aire caliente. Esto se hace seleccionando el ícono de **Path** de la barra de herramientas principal. Una vez haya hecho clic en el ícono de **Path**, use el botón izquierdo del ratón para adicionar nodos, los cuales actúan como puntos de ángulo para conectar la secuencia de rutas. Simplemente haga clic en el botón izquierdo del ratón cada vez



que se necesite un nodo en la ruta. Hacer clic en el botón derecho del ratón sobre un nodo trae más opciones—permitiendo la eliminación de un nodo, la adición de más nodos a la secuencia, el cambio de altura de una ruta, y el cambio de la ruta a un tipo particular de ruta, por ejemplo, un camino, una pared, una plataforma, etc.

Reflexión

Hacia el final de la clase, pida a los estudiantes que salven y apaguen sus pantallas por unos minutos para discutir lo que ellos encontraron durante su lección. Vea si ellos tuvieron problemas similares y si se encontraron soluciones. Discuta cómo resolvieron ellos los problemas—algunas veces esto acarrea prueba y error, otras veces esto incluye mirar código en los otros programas. La actividad tiene el propósito de que los estudiantes compartan estrategias.

Enlace a Campo de Actividad – Ciencias

La característica de ruta y los mosaicos de codificación “**Move /Toward**” (Mover/Hacia) y “**Fast/Slow**” (Rápido/Lento) en **Kodu** podrían ser particularmente útiles en la animación de acontecimientos biológicos y físicos.

Considere, por ejemplo, animar las tribulaciones de pez o tortuga durante sus rutas migratorias desde el mar (esquivando escombros y barcos pesqueros) hasta esclusas y escaleras de peces hasta depredadores naturales. Conceptos como sobrepesca podrían fácilmente ser puestos en un formato de juego también.

Desde una perspectiva de ciencia física, Kodu podría animar patrones estelares, trayectorias o formación de átomos. Animar estos conceptos puede ayudar a los estudiantes a retener conceptos con mayor profundidad y complejidad.



Hoja de Estudiantes – Actividad 3: Comportamiento del Objeto y Rutas

Objetivos: Los estudiantes mueven los personajes usando el teclado y ellos mismos programan comportamiento del objeto, crean una ruta que seguirá un personaje/elemento.

Direcciones: Acabamos de ver las herramientas para hacer mover a los personajes. Ahora es su turno para intentar hacerlo.

A medida que complete cada uno de los siguientes puntos, márkelo en su lista como realizado. Revise con su instructor una vez haya terminado o si está teniendo problemas.

Lista de Verificación de cosas para hacer:

Ir al paisaje que usted creó durante la última sesión o a un juego existente en **Kodu**. En consideración de la simplicidad, escoja un mundo que tenga tierra de algún tipo.

Crear un personaje que controle el usuario con el controlador.

Crear un segundo personaje que tenga movimiento automático.

Crear una ruta en la cual se mueva un tercer personaje.

Crear un objeto que haga algo ya sea cuando sea chocado, ve o se le dice que haga algo usando el controlador o cuando es programado para hacer algo automáticamente a través de una frase “**DO**”.

Actividad de Desafío

Después de que haya terminado sus cosas por hacer para la Actividad 3, pruebe en el **Tutorial 02**. Vea si puede descubrir cómo programar la motocicleta para moverse y disparar usando el teclado y ratón de acuerdo con las instrucciones.

Incluso dé un vistazo a “**3D Flare Paths**” para ver cómo hacer representación en 3D y un interesante sentimiento gráfico.

Revise el juego de acción **Rock Fight v09** para ver un ejemplo de rutas elevadas. Si usted quiere jugar el juego, necesitará convertirlo a una versión de juego para teclado. Asegúrese de cambiar la pantalla de entrada para el usuario, dirigiéndolo sobre qué teclas usar cuando juegue el juego. Usted puede cambiar esta pantalla yendo al ícono “**Wrench**” (Llave de Tuercas) en la barra de herramientas principal y cambiar el **Start Game With World Description** (Iniciar Juego con Descripción del Mundo). Además, cuando esté jugando y analizando el juego, piense sobre cómo se usan la ruta/las rampas y adicionan a la intensidad del juego.

Sesión 4: Hacer Clones y “Creatables” (Seres que se pueden crear)

Cuando hayan terminado, los estudiantes podrán:

- Crear un protagonista (controlado por el usuario), un antagonista (automático), y personajes periféricos
- Basar comportamientos y acciones del personaje en entornos y reacciones entre ellos
- Comenzar a entender el argumento de juego y la historia de fondo

Ahora que los estudiantes han creado configuraciones y han aprendido muchas de las características más sofisticadas de **Kodu**, ellos tratarán de adicionar personajes que interactúen y construyan argumentos. Hay una variedad de personajes en **Kodu**, y las interacciones entre ellos pueden tomar millones de diferentes pistas—todo depende de la imaginación del usuario. Algunos estudiantes estarán altamente influenciados por los géneros de juegos que ellos juegan, entonces usted tendrá niños construyendo juegos de acción de alto impacto mientras otros toman un enfoque más parecido a simulador. Sin embargo, es importante tener una mini lección sobre cómo interactúan los personajes y para ellos hacer una lluvia de ideas de los tipos de personajes que quieren colocar en sus mundos. Dada la profundidad de su interacción con **Kodu**, es probable que ellos ya hayan comenzado a construir historias con personajes, entonces esta actividad podría simplemente adicionar una tendencia meta-cognoscitiva sobre sus creaciones.

Comience preguntando a los estudiantes sobre sus libros favoritos mientras se definen algunos términos literarios. Por ejemplo, dependiendo de la edad de sus estudiantes revisar el significado de:

- Personaje principal (protagonista)
- Personaje opositor (antagonista)
- Personajes secundarios (periféricos)
- Argumento—el incidente gancho/que incita, acción que levanta, y clímax (lo que en términos literarios consiste en exposición, acción que levanta, clímax, resolución y conclusión.)

Después o mientras usted esté hablando acerca de cada uno de los términos anteriores, proyectar “**Turtle and Fish**” (Tortuga y Pez) en la pantalla para que los estudiantes identifiquen al personaje principal, el personaje opositor y personajes periféricos. Esto puede ser particularmente divertido para estudiantes si uno de ellos está a cargo del controlador. Otro enfoque para hablar acerca de los elementos literarios podría ser que el estudiante juegue

“Turtle and Fish” (Tortuga y Pez) en grupos pequeños con una discusión de estos elementos después de todo.

Ver si los estudiantes podrían identificar un argumento y una historia con antecedentes. Estas últimas dos características literarias serán un poco menos tangibles para los estudiantes dado que las estructuras del argumento varían— los argumentos de juego frecuentemente son planos haciendo que los variados componentes del argumento sean difíciles de identificar.

Los elementos son identificados como tales:

Protagonista Melvin la Tortuga

Antagonista Pez Volador Eléctrico

Personajes Secundarios Pez y Submarino

El argumento de la historia:

Exposición Pregunte a los estudiantes acerca de la historia antes del juego— ¿Cómo es la configuración de la historia? ¿Cómo y por qué están los personajes en oposición entre ellos? ¿Qué motiva a los personajes?

Acción Creciente ¿Qué mueve la historia hacia adelante? ¿Qué eventos mueven el juego hacia adelante?

Clímax ¿Cuál parece ser el clímax en este juego?

Resolución ¿Proporciona el juego una resolución o conclusión? Y si no, ¿cómo podría?

Conclusión No hay una conclusión, pero los estudiantes pueden hacer lluvia de ideas acerca de cómo podría terminar el juego.

Reflexión

Durante los últimos 20 minutos de clase, haga que los estudiantes compartan sus mundos de nuevo. Esta vez, quizás, hágalos compartir como clase completa. Si es posible, proyecte juegos en la pantalla y vea si los estudiantes pueden encontrar colectivamente los comienzos del antagonista, protagonista, y personajes periféricos. Los comienzos de la estructura de argumento pueden comenzar a emerger mediante estas discusiones. Una gran pregunta que podría ser hecha en este diálogo es cómo podría ser usado el puntaje para empujar la historia hacia adelante y agregar estrategia al juego—actualmente Melvin la Tortuga está sólo alimentando las rocas de regreso al pez y permaneciendo lejos del pez volador eléctrico.

Si terminan esta actividad antes, dígales a los estudiantes que se sientan libres de explorar los otros mundos que ya existen en **Kodu**.

Qué puede esperar usted de la Actividad de Estudiantes

La meta de esta actividad es que los estudiantes comiencen a pensar sobre sus juegos como más que simples entornos de carreras y disparos. Los juegos con frecuencia son una red compleja de interacciones, y aunque algunas historias son bastante planas en su estructura, la mayoría tiene personajes que están en oposición entre ellos, tienen alianzas, progresan a una meta particular. Empujar a los estudiantes a pensar sobre su trabajo como una forma de texto podría ayudarlos a diseñar mundos más complejos y código más enredado. Considere usar la actividad suplementaria para llevar adicionalmente a los estudiantes dentro de la creación de historias confusas.

Hoja del Estudiante Actividad 4: Personajes y Argumento

Objetivos: adicionar personajes y hacer lluvia de ideas y quizás comenzar implementando una estructura de argumento.

Direcciones: En lecciones anteriores, usted creó un paisaje. Si no ha agregado personajes aún a su paisaje, ahora es el momento para hacerlo. Antes de que comience a agregar o revisar sus personajes, piense un poco sobre la manera en que los personajes interactúan.

*Complete la lista de cosas por hacer y discuta con sus compañeros de clase las ideas que usted tenga. (Siéntase libre de consultar a **Kodu** para revisar sus mosaicos de robot en búsqueda de ideas de personajes.)*

Lista de verificación de cosas por hacer:

Lluvia de ideas de personajes

Protagonista (Personaje Principal)

- ¿Quién o qué es su personaje principal?
- ¿Qué motiva al personaje?
- ¿Cuáles son sus gustos y disgustos?
- ¿Tiene emociones?
- ¿Cómo se comunica con los otros?

Antagonista (Personaje Opositor)

- ¿Quién o qué es su antagonista?
- ¿Qué motiva al personaje?
- ¿Cuáles son sus gustos y disgustos?
- ¿Qué son sus emociones?
- ¿Cómo responde al personaje principal y por qué?

Personajes Periféricos (personajes secundarios)

- ¿Qué o quiénes son los personajes secundarios?
- ¿Qué rol juegan ellos—apoyan ellos al personaje principal o al antagonista?

En seguida de que usted haya respondido las preguntas anteriores, comiencen a poner a los personajes dentro del mundo que usted comenzó a crear antes. Si la configuración que usted hizo no parece estar de acuerdo con sus personajes, revísela o escoja un paisaje nuevo para alterar o construir.

Actividad de Desafío

Dar a un personaje una forma de hablar a través de una caja de diálogo—estas son como las burbujas en las caricaturas en las cuales

las personajes dicen algo. Esto se encuentra bajo el “**Say**” en los mosaicos de programación. ¿Cómo podría usar esta característica en forma eficaz en un juego?

Actividad Suplementaria

Si sus estudiantes están encontrando difícil crear una idea para sus mundos de juego, considere proporcionarles recordatorios/avisos creativos. A continuación hay un conjunto de primeras líneas (algunas adaptadas) de la literatura.

Semillas de Creación

Kodu, habiendo perdido su camino en un bosque tenebroso, y estando obstaculizado por bestias salvajes mientras asciende una montaña, se encuentra con un robot, quien le promete mostrarle tres mundos...

1. Había una vez.... Era una noche oscura y tempestuosa....

Kodu estaba empezando a sentirse muy cansado de sentarse al lado del río con su hermana, entonces....

Este es **Kodu**. Él es una pequeña criatura buena. Y siempre muy curiosa...

Después de un viaje de siete años, **los Koduianos** levantan velas para ir a casa, pero son tomados por una tormenta terrible. La tormenta hunde todos los barcos, excepto uno. Su líder exige que la tormenta pare, y las aguas se calman. Pero los vientos los han llevado fuera de su curso a una tierra de amigos y enemigos.

Kodu, obsesionado con viajar a través del tiempo, se construye una máquina del tiempo y, para su gran sorpresa, viaja más de 800.000 años al futuro. El mundo ha sido transformado por una sociedad que vive en una aparente armonía y felicidad. Pero cuando **Kodu** permanece en este mundo del futuro, él descubre un mal oculto....

En el último atardecer rojo, una línea negra de colinas bajas apareció en la distancia. Vi una criatura en la distancia con sus dos...

Los pobladores de la villa de **Little Leatonia** todavía la llamaban “**la Casa de los Riddle**” aunque habían pasado muchos años desde que la familia **Riddle** había vivido allá. Estaba erguida en una colina que dominaba la villa, pero no lucía tan grande como lo hizo una vez.

~~Todos los **Leatonianos** pensaban que la casa era tétrica pues hacía medio siglo algo extraño y horrible sucedió allá.~~

Durante la totalidad de un día aburrido, oscuro e insondable en el otoño, cuando las nubes colgaban bajas en el firmamento, yo había estado pasando por una parte deprimente del país. Repentinamente, me encontré a la vista de la extraña **Casa de Unsure**.

“Ahora,” dijo el rey, “tengo fe en esta búsqueda. Caballeros del la Guardia Azul partan. *Estoy seguro de que nunca los veré a todos ustedes juntos de nuevo. Salgan ahora a las praderas de **Galahad** para salvar a nuestro pueblo de la fatalidad. Descansen esta noche pues una larga jornada llena de criaturas y lugares maravillosos y extraños está en su futuro.”*



Nombre:

Direcciones: Complete las siguientes preguntas y luego presente sus pensamientos.

Configuración (¿a qué se parece el paisaje?)

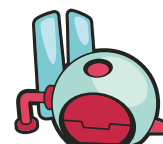
Modo (voluble, oscuro/ caviloso, ¿el paso del juego sube y baja?)

Personajes (comportamientos, conflictos, amigos, ¿alianzas?)

Objetos (¿Los árboles, edificios, rocas, etc., tienen una función especial?)

Argumento (¿Cómo progresa la historia?)

¿Por qué querría alguien jugar este juego? ¿Qué lo hará único?



Sesión 5: Comenzar Historias y Personajes Únicos

Cuando terminen esta sesión, los estudiantes:

- Estarán más a tono con la elaboración de estrategias
- Entenderán la influencia de modo y tono sobre el juego

Aunque los estudiantes probablemente han estado implementando diseños estratégicos en sus juegos, así como ejecutando estrategias a medida que ellos juegan los juegos de los otros, ellos deben ser concientizados de este elemento de juego. Estrategia es el enfoque que toma el jugador para ganar el juego, y la estrategia de programación está diseñada por el autor del juego para permitir que haya varias formas para que el juego sea ganado o perdido. En cualquier juego, probablemente hay varias formas para que el juego sea ganado.

Explíqueles que muchas de las estrategias que incluyó el diseñador del juego tenían el propósito de hacer que el juego fuera desafiante e interesante. Usualmente hay muchas formas de jugar un juego, y aunque la suerte frecuentemente juega un rol en ganar o perder, la estrategia requiere pensar a través del mejor método para una meta final. Entre más pueda el diseñador del juego anticipar y construir las estrategias y proporcionar ajustes para hacer que la acción sea más intensa o menos intensa, más exitoso podrá ser su juego. Cuando se esté estableciendo una estrategia en un juego, el diseñador podría tener en cuenta las habilidades del avatar del jugador, el entorno, la tarea, y el protagonista y personajes periféricos. Hable con sus estudiantes acerca de las diferentes formas en que la estrategia podría ser construida dentro de un juego. Use sus proyectos pasados y otros juegos a los cuales ellos hayan estado expuestos durante el curso para hablar sobre estrategia.

Modo y Tono

Modo y tono puede agregar mucho a la experiencia del usuario. Los medios y configuración en Kodu pueden cambiar el modo y tono en forma drástica. Si los estudiantes no han explorado estas características, usted debe señalárselas. Un cambio en la configuración puede cambiar la dificultad e intensidad de un juego una buena cantidad. Bajo el **Setting Tile** (mosaico de configuración), usted puede cambiar el cielo, iluminación, agua, y otros numerosos efectos. De los juegos varios que han explorado los estudiantes, pregúnteles cómo la configuración afectó el modo. Pregúnteles cómo el sonido y la música dentro de los juegos influyó la sensación del juego. A medida que los estudiantes hablan acerca de varios juegos, haga mención de los entornos para establecer contrastes que estimularán el diálogo adicional. Muestre a los estudiantes cómo cambiar la configuración del entorno, así como sonido y música. Esto se puede hacer mirando el código.



Hoja del Estudiante Actividad 5: Modo y Tono

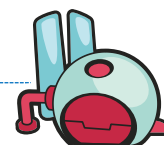
Objetivos: Revisar modo y tono con base en la estrategia

Direcciones: En clase ustedes han estado creando sus propios mundos de juego. Estrategia, intensidad, modo y tono siempre pueden ser mejorados.

Dado lo que ustedes han hablado en clase, piensen acerca de cómo puede ser mejorado su juego y cómo podrían ser cambiados el tono y modo.

Cosas para considerar:

- ¿Cree usted que el jugador tiene una posibilidad justa en el juego—es ganable por habilidad o por suerte? Si es más por suerte, entonces usted podrá querer revisar el juego.
- ¿Cuáles son las estrategias que puede usar un jugador para abrirse paso a través de su juego?
- ¿La acción del juego lo hace entrar al mundo de los juegos?
- ¿Se usa la música o los efectos de sonido en el juego? ¿Son ellos extremadamente molestos o sí crean tono o modo que cautive al jugador?



Sesión 6: Estrategia, Modo y Tono

Cuando terminen esta sesión, los estudiantes podrán:

- Entender la clonación y los “**creatables**” (seres que pueden ser creados)

Esta lección se beneficiará de los contrastes que hay entre el copiado y los **Creatables**, y utilizará la frustración como una ventana clave dentro del aprendizaje. Crear ejércitos, grupos, tesoros escondidos, manadas, enjambres/bandadas, etc., es fácil en **Kodu**. Sin embargo, dar a los individuos que están dentro de los grupos el mismo conjunto de cualidades no es fácilmente intuitivo y puede llevar mucho tiempo cuando se utilice solamente la característica de clonado. Usualmente, los estudiantes comienzan clonando (copiando personajes) y luego pronto se cansan de programar los mismos atributos del personaje múltiples veces si ellos quieren editar el comportamiento del grupo. Los **Creatables** permiten al usuario hacer el cambio una vez, en vez de múltiples veces como uno tendría que hacer si simplemente hicieran copias del personaje.

La lección en clase hará que ellos usen la característica de “**clonar**” para los dirigibles y luego traten de modificar el comportamiento del grupo—esto no se hace fácilmente porque para los cinco dirigibles que ellos hacen, ellos necesitarán cambiar cinco líneas de código diferentes. Como una respuesta a esta frustración, luego usted los llevará por cómo usar los **Creatables** en los cuales el código puede ser cambiado solamente una vez. Haga que los estudiantes manejen los controles a medida que usted habla sobre qué hacer. Con toda la clase o en grupos, haga que los estudiantes sigan los pasos mencionados a continuación.

Clonado y **Creatables**

Vaya a **Dirigibles y Jets**. Usted notará que hay un dirigible en un lado del campo y un jet en el otro. La tarea de la clase o grupos pequeños es hacer una batalla en la cual el jugador controla ya sea los dirigibles o los jets.

Primero, programe el dirigible. Usted notará dos líneas de código para el dirigible. La música dramática es codificada para cuando el dirigible vea el jet a la distancia, y el dirigible brilla cuando se coloca cerca al jet. Programe el dirigible de forma tal que se mueva usando el teclado.

Posteriormente de que haya revisado el código para el dirigible, haga hasta cinco clones/copias de él usando el botón derecho del ratón y seleccionando “**Copy**” (Copiar). Ponga el cursor en la posición donde usted quiere que vaya la copia del dirigible y luego haga clic de nuevo con el botón derecho del ratón y escoja **Paste** (Pegar) (dirigible). Haga cuatro dirigibles mas, usando el método de **Copy** y **Paste** (Copiar y Pegar).

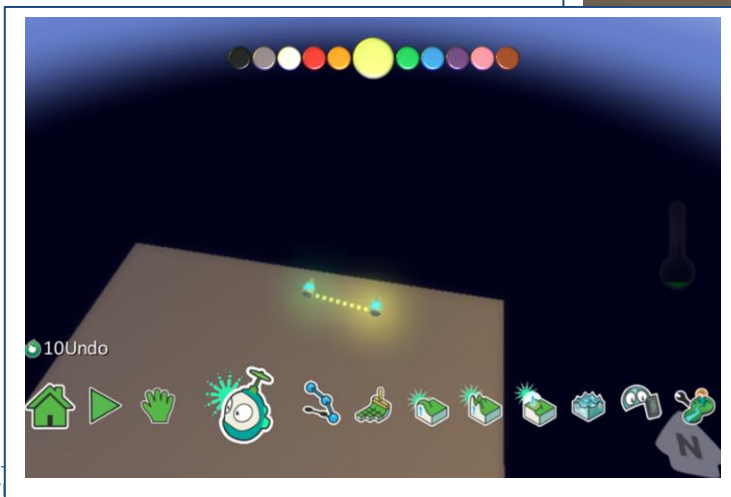
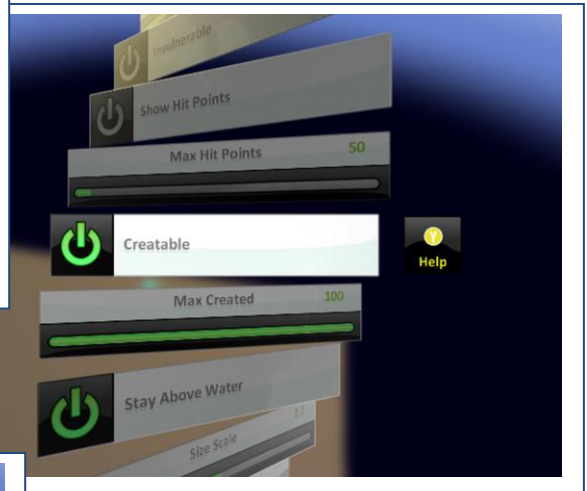
Ahora, asegúrese de que los dirigibles puedan moverse hacia adelante.

Ahora dé a los dirigibles la habilidad de disparar “**luces o señales**”. ¿Cuántas veces necesita cambiar el código?

Pregunte a la clase si hay alguna forma en la cual hacer para que usted pueda simplemente ingresar código en una ubicación. La respuesta es **Creatables**.

Demuestre esto haciendo un segundo conjunto de dirigibles que operen automáticamente sin la contribución del usuario. Haga que el estudiante coloque el nuevo conjunto de dirigibles en algún otro lugar. Haga que el dirigible sea de color verde u otro color para distinguirlo del primer conjunto. Haga que brille en un color diferente también.

Una vez que el nuevo dirigible sea creado, haga clic con el botón derecho del ratón y seleccione “**Change Settings**” (Cambiar configuración). Encuentre **Creatable** en la lista, y prenda la disposición. **Settings** y desplácese hacia debajo de la lista a **Creatables**. Seleccione esta opción. Luego, vuelva al dirigible y cópielo. Si usted coloca el cursor sobre uno de los dirigibles, aparecerá una línea punteada entre los dos dirigibles, indicando que los dirigibles son parte de una cadena de *creatables*.



Siéntase libre de copiar más dirigibles del dirigible principal, lo cual crea más versiones *creatable* del mismo.

Por último, agregue código al dirigible “principal”, por ejemplo, “***shoot blips at jet when close by***” (disparar luces o señales al jet cuando esté cerca). Muestre a los estudiantes cómo se transfiere este código a todos los dirigibles. Cuando usted juegue el juego real, el ***Creatable*** principal no estará en la pantalla—sólo aparece durante el modo de edición.

Anote a los estudiantes que ellos no deben agregar demasiados personajes a sus juegos ya que eso puede colocar al programa en peligro de hacerse pedazos.

Enlace a Campo de Actividad— Estudios Sociales y Ciencias

El uso de ***Creatables*** es poderoso y puede crear un efecto impresionante de una legión de tropas todos moviéndose en la misma forma. Considere cómo podría ser usada la herramienta para diseñar las acciones de una batalla famosa mientras se usan las fuentes primaria y secundaria como la base de investigación.

De forma interesante, ***Kodu*** de la misma manera, puede ser usado para crear una apariencia más naturalista. Por ejemplo, piense acerca de ***cardúmenes*** de peces donde cada ***cardumen*** es su propio conjunto de ***Creatables*** programado para preguntarse. Luego dentro de cada conjunto, usted puede programar un conjunto de atributos únicos para el cardumen de peces.

Hoja del Estudiante Actividad 6: *Creatables* (seres que se pueden crear)

Objetivos: Hacer clones y entender la premisa de *Creatables*.

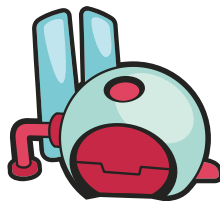
Direcciones: Siga la lista de cosas por hacer a continuación y haga que su instructor revise su progreso antes de moverse a la actividad extra. Esta es una actividad bastante abierta, de forma tal que siéntase libre de ser creativo en su uso de los ***Creatables***.

Lista de verificación de cosas por hacer:

En un mundo que usted ya ha diseñado o en un juego nuevo en general, agregue dos o tres conjuntos de personajes, como lo hicimos con los dos conjuntos de dirigibles y el conjunto de jets. (**Los *Creatables*** no necesariamente necesitan marchar al unísono. Use ***Move*** (Mover), ***Wonder*** (Preguntarse) para codificar un sentimiento más naturalista.)

Actividad Extra

Si usted termina y tiene algún tiempo de sobra, de un vistazo a “***Technique*”: *Launching Creatables v02***” (Técnica: Lanzando *Creatables* v02). Note cómo tiene que ser convertida la manzana en un *creatable* para que **Kodu la lance**. Llévase la manzana lejos, y **Kodu** no podrá lanzar nada. ¿Puede usted razonar en cuanto a por qué sería esto?



Sesión 7: Cambiar Comportamientos Usar Páginas, Establecer, y Cambiar Perspectivas

Cuando terminen esta sesión, los estudiantes podrán:

- Entender la idea de páginas, cuándo son usadas y por qué
- Usar características cercanas y lejanas
- Cambiar ángulos de la cámara en la configuración y en el código

Esta lección está basada en el descubrimiento. Haga que los estudiantes jueguen **Vendura v14**. Después de que ellos hayan jugado el juego un buen rato, pídales mirar el mundo y ver si ellos pueden determinar cuántos objetos y personajes están incluidos en el mundo y qué hace cada objeto o personaje.

Aquí está una lista:

Turtle (Tortuga) — el usuario controla, adquiere habilidad de disparo de espirales por interacción con **Kodu**

Stick (Palo) — informa a la tortuga (el usuario) que **Koduwan** ha estado esperando

Kodu — infunde poder y desaparece

Wisps (Espirales) — dados a la tortuga como un poder

Castles (Castillos) — desencadenan minas que disparan

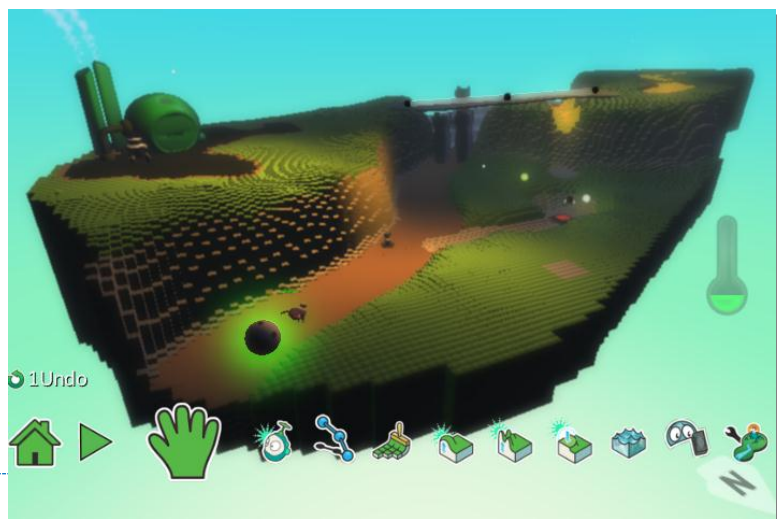
Hearts (Corazones) — *creatable* que actúa como comida

Mines (Minas) — se mueven aleatoriamente y disparan a **Kodu**

Factory (Fábrica) — actúa como el objeto que desata el final

Pushpad—informa a la tortuga qué hacer

Si los estudiantes no han apuntado al código, asígneles un objeto o personaje para evaluar. A ellos se les debería encomendar la tarea de reportar de regreso al grupo cómo es codificado el objeto o personaje. Usando el proyector, haga que ellos presenten cada uno de sus hallazgos. Durante su presentación, ponga atención



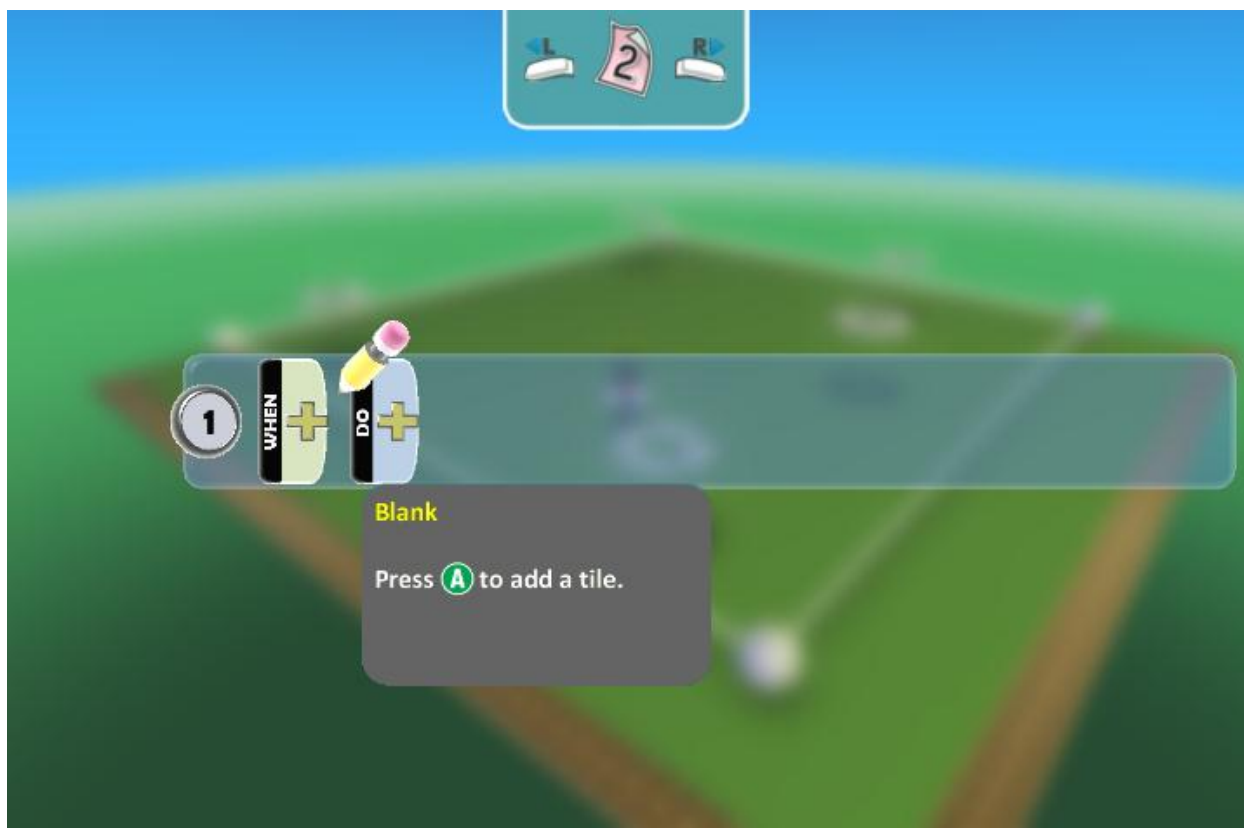
particular a las páginas, y en forma secundaria a cómo se usan el *nearby* (cercano), *far away* (lejano), y **camera shifting** (alternación de cámara).

Para ayudar a guiar la discusión, considere cómo hacen en **Vendura**, los objetos y personajes para proporcionar al jugador un entendimiento de lo que está sucediendo. Ellos actúan como direcciones para qué hacer a continuación. Sin embargo, para crear el efecto de acción de alternación, son necesarias las páginas. Las páginas ayudan a alternar los comportamientos ya sean basados en tiempo o como reacción a alguna ocurrencia.

Páginas

El mando que dirige a la tortuga a **Koduwan** es el primer uso de las páginas. El código cambia el estado del mando desde cerrado (página 1), el cual es su estado constante, a abierto (página 2). La página 2 entonces expresa otro conjunto de comportamientos los cuales corre y luego retorna al estado inicial de cerrado (página 1) después de cierta cantidad de tiempo. El segundo uso de las páginas viene con la aproximación de **Koduwan** por parte de la tortuga. La página 1 proporciona un estado de descanso para el **Kodu**. La página 2 activa un comportamiento, específicamente **Koduwan** volteándose y saludando a la tortuga. Inmediatamente de tres segundos, la Página 3 es alistada. Ella proporciona otro conjunto de texto para hacer que siga historia. En seguida de ocho segundos, se crea un “**espiral**” y le es dado a la tortuga a medida que **Koduwan** desaparece en una explosión.

Esta es una demostración de cómo funcionan las Páginas—ellas permiten un cambio en el comportamiento del personaje. Con frecuencia el cambio es desatado por un evento dentro del juego. Por ejemplo, comer una manzana puede permitir que un personaje que en forma típica es perseguido por otro personaje que en forma repentina sea el agresor. Este cambio en comportamiento puede ser temporal o permanente dependiendo de cómo sea codificado el programa. Se pueden agregar Páginas ingresando a **Program**, y luego alternando a una página diferente ya sea haciendo clic en **L** ó **R** en la parte superior de la ventana de **Kodu**.



“Close-By” (Cercano) y “Far Away” (Lejano)

Se debe señalar a los estudiantes qué tan dramático es el efecto creado a través de esta serie de intercambios y comportamientos. Aunque las páginas ayudan a facilitar las operaciones, los mosaicos de “*close-by*” y “*far away*” permiten que los usuarios creen un elemento de sorpresa. En la secuencia de ingreso, tanto el palo como el comportamiento de **Koduwan** son guiados sintiendo a la tortuga cerca y luego se apagan con base en el cronómetro. Esto es eficaz para hacer que la historia siga adelante. Este tipo de uso eficaz debe ser mencionado a sus estudiantes. Cambio de Cámara

Siempre, puntualice cómo cambia la perspectiva de la cámara cuando la tortuga flota sobre la plaza de tierra roja. Cuando se está jugando, este cambio en la perspectiva parece ser causado por la aproximación de la tortuga a **Koduwan**. Este es un uso eficaz de la tierra dado que la cámara sacude a los jugadores dentro de una perspectiva diferente. El posicionamiento de la cámara

Tipos de perspectiva del Jugador

Primera Persona – Los jugadores ven el juego a través de los ojos del personaje.

Tercera Persona – Los jugadores existen dentro del mundo de juego lo cual produce menos inmersión.

Top-Down (de arriba hacia abajo) – el jugador tiene una visión global del mundo de juego

“Side-scrolling” (Desplazamiento lateral) –Paso rápido; la acción es vista desde una cámara de visión lateral; no muestra mucho acerca del mundo.

es una herramienta eficaz para la creación de un efecto particular. Quizás haga que los estudiantes piensen acerca de la diferencia en efecto entre la primera y la tercera persona — ¿Cuándo y por qué usaría uno u otra? Aunque la cámara puede ser cambiada usando los mosaicos, igualmente puede ser cambiada bajo configuración en el menú principal.

Actividad Extra

Haga que los estudiantes restablezcan la perspectiva original.

En la Página 2 escriba:

When: see, Kodu, none, close by Do: Switch Page 3 (Cuando: ver, Kodu, ninguno, cerca Hacer: Cambiar Página 3)

En la Página 3 escriba:

When: see, Kodu, none, nearby Do: open, once (Cuando: ver, Kodu, ninguno, cerca Hacer: abrir, una vez)

When: see Kodu none Do: switch, page 1 (Cuando: ver Kodu ninguno Hacer: Cambiar página 1)

Introducción al Proyecto de Juego

Cierre la sesión hablando acerca del proyecto final para **Kodu**. Los criterios para el proyecto dependen de usted, pero al introducir el proyecto ahora, los estudiantes tendrán bastante tiempo para generar ideas y para trabajar en sus proyectos. Asegúrese de hacer que los estudiantes planeen sacando o planeando sus juegos a través de una web de lluvia de ideas, escritura libre narrativa, o algún otro formato para hacer que fluyan los jugos creativos



Hoja del Estudiante Actividad 7: Ángulos de Cámara y Cambiar Comportamiento

Objetivos: cambiar la perspectiva de la cámara, usar “**close-by**” (cercano) y “**far away**” (lejano), implementar el cambio en comportamiento usando páginas.

Direcciones: Usted acaba de dar un vistazo a código y técnicas para crear efectos dramáticos y para elaborar cambios en comportamiento dentro de un personaje. Ahora es el momento de practicar.

Lista de verificación de cosas por hacer:

Considere las tres cosas sobre las que usted habló durante la lección de hoy—**close-by** (cercano) y **far away** (lejano), ángulos de cámaras y páginas.

Explore los varios juegos que usted tiene en su plataforma de **Kodu** y cómo podría usted implementar las tres operaciones dentro de un solo juego. Por ejemplo, mire en **Chaotic Orbitals v3** y entre un código de cambio de cámara a primera persona, quizás manteniendo presionada la barra espaciadora. El cambio de cámara también podría ser iniciado mediante un choque, llegar cerca a, comer, o agarrar la moneda.

Ahora, agregue personajes que cambien comportamientos basados en sus interacciones entre ellos o con objetos en el mundo. Para crear este sistema de interacciones, usted puede necesitar tiempo para hacer una lluvia de ideas y jugar con los mosaicos para ver qué hay disponible para usted. Si necesita ayuda, considere agregar personaje y objetos en **Chaotic Orbitals v3** que tengan un estado de descanso (Página 1) y luego son estimulados a otro comportamiento con base en interacciones con otros personajes.

Actividad Extra

Si usted termina la actividad anterior y quiere otro desafío de programa, vaya a “**Technique, Change Behavior**” (Técnica, Cambiar Comportamiento). Primero juegue el juego y vea si puede “ganar” el juego. Luego, programe el juego de forma tal que usted vea el juego desde la perspectiva del **cycle**.

O

Trabaje en su propio mundo de juego.

Sesión 8: *Power Ups* (aumento de poder), *Health* (Salud), y *Timer* (Cronómetro)

Cuando hayan terminado esta sesión, los estudiantes podrán:

- Usar cronómetros, monitores de salud, y "**power ups**"

En los juegos que hemos usado en las lecciones y que usted probablemente ha pedido a los estudiantes que exploren independientemente, usted puede haber visto cronómetros, monitores de salud y **power ups** siendo usados. De hecho, algunos estudiantes ya podrían estar usando estas características en su juego y diseño de juegos. Esta sesión mirará estas tres áreas de una forma un poco más cercana.

Para comenzar, discuta con la clase el concepto de **power ups**. Con toda probabilidad, ellos tendrán una idea bastante buena de la forma en que ellos operan. Básicamente, un personaje come, agarra, choca, etc., algo que le da una destreza o habilidad especial para ventaja en el juego—con frecuencia esto es temporal. También hay "**power downs**" (disminución de poder) los cuales tienen el efecto opuesto. **Technique: Change Behavior** (Técnica: Cambiar Comportamiento) es un ejemplo de un **power up** muy sencillo—al comerse la manzana, el **robot (bot)** es capaz de saltar. Esto podría ser demostrado en el proyector. Algunos estudiantes podrían ya haber experimentado con este juego.

Ahora, haga que los estudiantes exploren dos juegos que utilizan "**power ups**", al mismo tiempo ellos también se están conectando a puntajes y salud. Diga a los estudiantes que usted quiere que ellos jueguen cada uno de los juegos e identifiquen el **power up** (o **power down**) y todavía identifiquen el código asociado con cada uno. Después de que ellos hayan jugado los juegos, haga que los estudiantes tomen turnos demostrando y hablando acerca de cómo funciona el **power up** en el juego.

De igual forma, es importante que los muchachos comiencen a pensar en forma crítica acerca de los juegos. Por ejemplo, estimúlelos a representar un diálogo del tipo "**Siskel y Ebert**" para ayudar a facilitar la discusión. Sería fructífero para ellos hablar acerca de cómo podrían ser implementados de una mejor forma los **power ups/downs** y se presten para el juego estratégico. Si ellos no entran en esta discusión sin demora, considere preguntarles cómo podrían jugar un rol los tipos de tierra, cronómetros, aliados, ladrones, etc. No están asociados poderes especiales con la adquisición de las monedas, manzanas y corazones en estos juegos, lo cual da por sentado el cómo los **power ups y downs** podrían ser implementados de mejor forma.

Los monitores de salud si juegan un rol en los juegos. Pregúnteles cómo estos agregan un segundo elemento estratégico más allá de la simple asignación de puntajes.

Los dos juegos son:

Rock Fight KB—el jugador necesita agarrar corazones como munición y pierde puntos al disparar; el monitor de salud sube a medida que son **agarrados** los corazones y baja a medida que se libera munición. Use las flechas para moverse y la tecla izquierda de mayúsculas y “Z” para lanzar rocas.

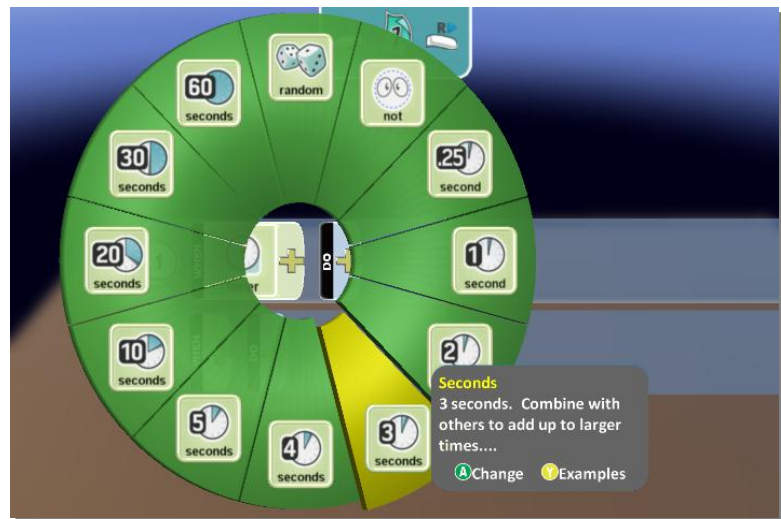
Pandemica KB—el jugador necesita comer manzanas para acumular puntos y ser sanado.

Estableciendo Código para el Cronómetro, Monitor de Salud, y Puntos

Estas tres herramientas pueden aumentar la comunicación dentro del juego señalando la condición de los personajes, así como mostrar cómo se están desempeñando ellos en el mundo del juego. Obviamente, los puntos son cruciales para ganar o perder un juego, pero también ellos pueden ser usados para facilitar la toma o retiro de poderes o habilidades. Todas estas herramientas pueden ser encontradas en la rueda de Mosaico principal bajo **Program** (Programa).

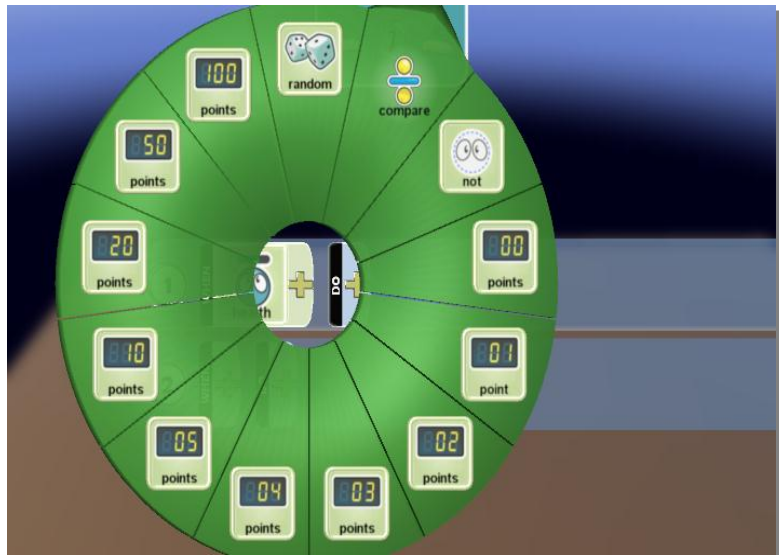
Timer (Cronómetro)

El **Timer** puede ser usado en innumerables formas—superando dentro del tiempo los escenarios de juego para establecer cuanto tiempo un personaje/jugador tiene o no tiene municiones, un poder o una herramienta. Lo anterior puede ser facilitado trabajando entre Páginas (lección previa), en las cuales el código de comportamientos para un personaje/jugador está atado dentro de un límite de tiempo.



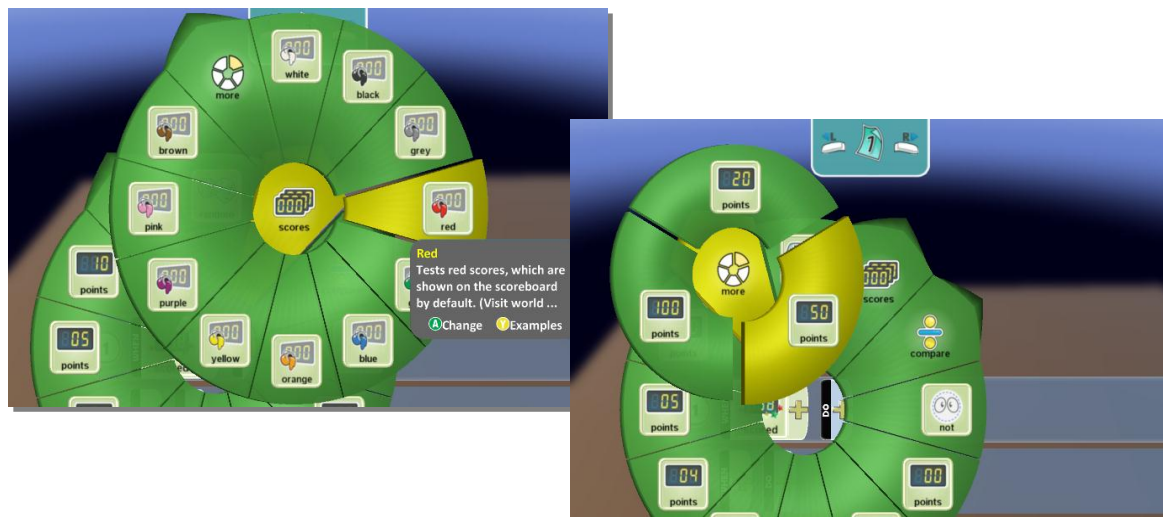
Health Monitor (Monitor de Salud)

El ***Health Monitor*** es una forma eficaz para comunicar información al jugador acerca de su personaje, así como los otros personajes en el campo de juego. Ellos siempre pueden facilitar cambios en el comportamiento. Por ejemplo, cuando la salud de un personaje cae por debajo o por encima de un límite de puntos el código podría cambiar a otro conjunto de comportamientos que empuja la narrativa del juego o juega hacia adelante. La salud del mismo modo, puede ser dispuesta en comparación con un total de puntos de conjunto o los puntos de salud de otro personaje.



Puntos Generales

Los sistemas de puntos de un mundo de juego pueden ser tan complejos o sencillos como los quiera hacer el usuario. Los contadores de puntos de personaje se distinguen mediante la designación de color en el código.



Creación de Juegos

Haga que los estudiantes continúen desarrollando sus mundos de juego. A medida que ellos comienzan a desarrollarlos, estimúelos a pensar sobre cómo podrían funcionar en su mundo la salud, cronómetros y sistemas de puntos. Los estudiantes no tienen que incluir todos estos elementos, especialmente si su juego está basado más en narrativa, pero ellos deberían comenzar a pensar sobre cómo uno de los elementos podría ser incorporado en sus mundos de juego.

Reflexión

Durante los últimos 20 minutos de su sesión, haga que los estudiantes compartan lo que programaron entre ellos. Si es posible, trate de anotar en forma verbal las diferentes formas en las cuales los estudiantes abordaron el problema. O trate de sacar a la superficie los problemas que estaban teniendo los estudiantes mientras ellos creaban sus “**power ups**”. La intención es compartir la solución de problemas. Haga que ellos hagan un diario sobre bien su propio juego o el juego de otro.

Enlace a Campo de Actividad:

Los expertos en matemáticas saben que todo es matemáticas — o al menos puede ser convertido en matemáticas. **Kodu** ofrece un número de formas en las cuales un instructor puede acentuar las matemáticas aplicadas mientras los estudiantes elaboran comportamientos y se comprometen en el diseño.

Aunque la codificación se presta naturalmente al concepto de ramificación, las características asociadas con salud, oportunidad, y puntos pueden colocar el razonamiento matemático en sobremarcha especialmente ellas se usan para facilitar una acción o comportamiento y comunicar información.

Incluidos dentro de las herramientas están sistemas de comparación, que coordinan soluciones de problemas mayores que y menores que.

- ¿Cree usted que el jugador tiene una posibilidad justa en el juego—es ganable por habilidad o por suerte? Si es más por suerte, entonces quizás usted quiera revisar el juego.
- ¿Cuáles son las estrategias que puede usar un jugador para salir adelante en su juego?
- ¿La acción del juego lo mete a usted dentro del mundo de juego?
- ¿Se usan en el juego música o efectos de sonido? ¿Son ellos excesivamente fastidiosos o crean tono o modo que cautiva al jugador?

Hoja del Estudiante Actividad 8: Cronómetros, Salud y “Power Ups”

Objetivos: Usar cronómetros, monitores de salud, y “*power ups*”.

Direcciones: Usted acaba de dar un vistazo a cómo funcionan los *power ups*. También ha identificado cómo son aplicados los monitores de salud a los personajes para activar otro nivel de estrategia dentro del juego. Ahora, es su turno para implementar los “*power ups*”.

Lista de verificación de cosas por hacer:

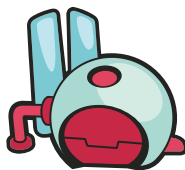
Use ya sea su propio mundo de juego o un mundo existente para crear un sistema de “*power ups*” y “*downs*”.

Mientras usted está considerando cómo iniciar su sistema, piense acerca de cómo podrían estar involucrados tierra, objetos y otros personajes.

Para aumentar una capa de desafío, trate de hacer que el “*power up*” o “*down*” sea temporal—esto quiere decir que lo coloque en un cronómetro.

Actividad Extra

A medida que usted y sus pares desarrollan juegos, jueguen y ofrézcanse realimentación entre ustedes. Su instructor le estará pidiendo que reflexione ya sea sobre su juego o el juego de otro y cómo es desarrollada la estrategia dentro del mismo.



Sesión 9: Más sobre Puntaje – Fundamentos para la Comunicación

Cuando hayan terminado esta sesión, los estudiantes podrán:

- Usar puntaje en formas más complejas

Sus estudiantes probablemente han estado usando puntaje mientras juegan y desarrollan sus propios juegos. Si ellos no han estado poniendo mucha atención al mismo como una herramienta para controlar comportamientos, entonces esta lección será beneficiosa para usted y sus estudiantes.

Cree un nuevo nivel para esta lección. Usted puede llamar a esta una Lección de Puntaje. Utilice un paisaje sencillo. Agregue un **Kodu** y al menos **4 cycles**. Programe a **Kodu** para que dispare siguiendo la orden. **Kodu** debe marcar 5 puntos cada vez que dispare a un **cycle**. Usted puede crear este código en frente de los estudiantes. Pregúnteles cuántos **cycles** necesitará agregar usted antes de que **Kodu** pueda ganar el juego (la respuesta es 4). Ahora agregue un enemigo para **Kodu** que lo pueda dañar de alguna manera. Agregue espirales que se muevan aleatoriamente y sustraigan puntos del puntaje de **Kodu**. Solicite un voluntario para agregar estos espirales. Ellos tendrán que cambiar el programa para **Kodu** de forma tal que este puntaje cambie cuando él sea golpeado por el espiral. Pídales que cambien el programa para los espirales de forma tal que ellos sean atraídos a **Kodu** una vez que él llegue a 15 puntos. Esto hará que el juego sea más emocionante.

Haga que los estudiantes jueguen **Bonk Out v.18** con sus grupos. Después de jugar este juego, discutan si el juego fue divertido. Haga que ellos miren al código y discutan cómo se usan el cronómetro y los puntos en el juego. Pregunte cómo se compara la acción del juego con otros juegos que también usan puntos, medidores de salud, y cronómetros. La clave para entender el puntaje es notar que no es solo un método para hacer seguimiento a quién gana y quién pierde, sino que todavía puede ser un método para ajustar comportamientos y para comunicar.

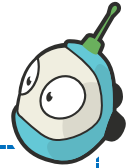
Creación del Juego

Los estudiantes gastarán el resto de la sesión trabajando en sus juegos para sus proyectos finales. Durante la última media hora de clase, haga que los estudiantes compartan su trabajo ofreciéndose realimentación entre ellos.

Reflexión de Diario

Termine la clase con un ejercicio corto de diario. ¿Creó usted su juego teniendo en mente una audiencia? ¿Qué tipos de gente cree usted disfrutaría más de su juego? ¿Está involucrado su sistema de puntos o un cronómetro en la construcción del juego? ¿Cómo son usados ellos? Basados en mirar a otros jugar su juego y en la realimentación que usted ha obtenido, ¿su juego es demasiado desafiante o fácil? ¿Qué lo hace pensar eso?

Hoja del Estudiante Actividad 9: Puntaje y Comportamiento



Objetivos: Use el puntaje para cambiar un comportamiento

Direcciones: Ahora usted verá cómo agregar el puntaje a su juego – usted puede usar diferentes puntajes de color par hacer seguimiento a cosas diferentes. Luego, haga que en su juego se tome una acción con base en el valor de un puntaje.

Cosas por hacer

Escoja cualquier entorno de juego que usted quiera para completar las siguientes tareas

Cree un juego sencillo en el cual se den puntos por hacer ciertas cosas—como comer, chocar, mantener/soltar, etc.

Diseñe una situación en la cual un competidor o bien sustrae puntos de su puntaje o tiene su propio contador de puntaje que incluye un color diferente

Codifique un sistema en el cual una acción es tomada o no tomada con base en un conjunto de puntajes.

Actividad Extra

Trabaje en su mundo de juego para su proyecto final.



Final de Kodu: Presentaciones de los Estudiantes

Presentaciones de los estudiantes

Toda esta sesión será dedicada a las presentaciones de los estudiantes. Personas externas (padres, desarrolladores de juegos en la comunidad, otros estudiantes) estarán invitados a venir y jugar los juegos que han creado los estudiantes. Esta es una gran oportunidad para que los estudiantes expongan y compartan los juegos que ellos trabajaron para crear.